RSTS-11 Sig
DECUS

DIGITGRAPH
P. O. Box 4249
Tucson, Arizona 85717
5/4/78

APRIL 1978 RSTS-11 SPECIAL INTEREST GROUP NEWSLETTER VOL 5. NO. 2

From the SIG Chairman

The schedule is set for Spring DECUS in Chicago. It is so crowded with RSTS sessions that we had to include night sessions. Stuart Katz has done an excellent job scheduling this Symposium. It is good to see more user led workshops and papers than in the past. That is the name of the game as RSTS matures toward a stable product. We must continue to be willing to share information among ourselves. Ask the people in the PDP8 world how important sharing becomes. Continuing in the same vain I want to thank Mike Mayfield for volunteering to write a series of excellent RSTS internal articles. This effort takes alot of time and I hope you will let Mike know how helpful it is to you.

As those of you who attended Fall DECUS know, microfiche copies of back issues of the RSTS newsletter were handed out. There are still 200 copies available free on a first come, first serve basis by writing the DECUS office.

See you in Chicago!

Bob Branton
RSTS SIG Chairman
3811 Farmville Drive, #194
Dallas, Texas 75234
(214) 243-0420

DECUS, DIGITAL EQUIPMENT CORPORATION, and the editor assume no responsibility or liability for articles or information appearing in this document.

Here is a list of the hard working people who are chairing RSTS Subgroups:

A. COMMERCIAL RSTS USERS
Stu Katz
Commercial Computer Services
1011 E. Tonky
Des Plaines, IL 60018
312-298-7766

EDUCATIONAL RSTS USERS
Howard Kipple
New Mexico Junior College
Hobbs, NM 88240
505-392-6526

CUSP EDITOR
Jon Spencer
PASCAL Computing
432 S. Barrington Ave.
Los Angeles, CA 90049

JSTATS, DSTATS DOCUMENTATION
Mike Mayfield
Data Processing Design, Inc.
6980 Aragon Circle, #7
Buena Park, CA 90620
714-994-4971

NEWSLETTER EDITOR
Murray Stinson
Scientific Placement, Inc.
2700 S. Post Oak Rd., #2430
Houston, TX 77056
713-621-9811

LARGE RSTS SYSTEMS GROUPS
Joan Salkin
U. S. Leasing
633 Battery St.
San Francisco, CA 94111
415-445-7400

TECHNICAL EDITOR
Carl Marbach
Acme-Hardesty Company
910 Fox Pavilion
Jenkintown, PA 19046
215-886-0200

B. WORKING GROUPS:

RSTS USER PROFILES
Bill Tabor
Florida Computer, Inc.
99 N. W. 183 St.
N. Miami, FL 33169
305-652-1710

RSTS DECUS LIBRARY PROGRAM
Ardoth Wilson
Central State University
100 N. University
Edmond, OK 73034
405-341-2980 Ext. 321

PROGRAMMING STANDARDS DEVELOPMENT
Roy Swatzell, Jr.
V. A. Hospital
Room 2 R 12
Birmingham, AL 35233
205-933-8103 Ext. 239

RSTS USER WISH LIST DEVELOPMENT
Bill Sconce
Industrial Specialties
P. O. Box 4228
Louisville, KY 40204
502-895-2914

TO: RSTS SIG

FROM: STUART KATZ
RSTS SIG SYMPOSIUM COORDINATOR

RE: DECUS SPRING SYMPOSIUM APRIL 25-28th 1978 IN CHICAGO

The symposium committee met in Maynard on February 16th and 17th to schedule the spring symposium and it was my pleasure to represent the RSTS SIG. Our task was to schedule the various sessions, trying to eliminate as many conflicts as possible. The late arrival of approximately 15 related sessions and the simple fact that there were approximately 45 RSTS sessions plus the EDUSIG, DEBUD, REST EDU SUB-SIG, and hardware sessions to work with. In keeping with the concept of providing time to all or those who submit sessions, we have had to make numerous cuts in the requested time slots and some coincidental scheduling of "bit and byte" type sessions versus the "application" type papers.

We have a RSTS gathering room available where "birds-of-a-feather" type sessions can be held. This will also be a place where lonely RSTS users can gravitate to in order to talk to other users. This room will be staffed by RSTS and solely available to us. We have also scheduled particular sessions (like Anton) for the last session of the afternoon so it can extend itself if need be. We hope that this type of scheduling will ease the agony of the ever present conflicts.

In San Diego, there was great concern due to the influx of VAX related sessions. Since there will be similar types of sessions. We have scheduled the second session on Tuesday (10 AM) to be a VAX for RSTS session. This is meant to provide you quick and concise information in reference to how VAX will effect you. Hopefully, you will be able to determine whether you need to schedule further VAX sessions into your four day schedule. This session will be given by Mark Bramhall, a long time RSTS developer and newly assigned VAX developer. Jim Miller will provide a similar type of overview for DECNET for RSTS/E directly following Mark's presentation.

Beside the special sessions, we will also hear from DEC on the following topics:

1. RSTS/E version 6C
2. RSTS/E version 7A and Future
3. BASIC PLUS to BASIC PLUS 2
4. RMS optimization
5. BASIC PLUS optimization

and

-more-

Users have been prolific with their participation and these include SUB-SIG sessions, "bit and byte" work shops and a range of application papers. These include:

1. Disk structure and optimization
2. Performance analysis and optimization

3. Wide range of applications
4. SUB-SIG sessions
5. Everything you wanted to know about SYSGEN (by popular demand)
6. The never ending "Wish List"

and

-more-

You will also find a super selection of RSTS EDU SUB-SIG sessions available. (We have done our best to eliminate conflicts for EDU RSTS users in particular.

We again have tried to arrange for a program exchange capability. At this time we don't have the details, but we feel confident that such a capability will be available. The best medium that we have a handle on today is 800 BPI, 9 track tape and we hope to be able to provide RK05 and/or RK06 type disks. We need volunteers for this exchange very badly. It is hard work but most worthwhile. Please help! (Write to me if you can help.)

As you can see the outline for the four days could easily consume a great deal more paper, so we are asking you to read the mini-papers when you receive your symposium package. You will find that most of the mini-papers give a thorough review of what they will cover. We have scheduled a session called "Welcome to DECUS for RSTS" as the first session on Tuesday. This session will act as a road map for RSTS users and we will try to eliminate any confusion regarding sessions at that time. We will also introduce a group of people that you may or may not know, but will want to be sure to meet. These will include SIG steering committee members, DEC staff and DECUS board members.

All and all, you can see that Chicago Spring Symposium will be a super four days. Now all we need is you! Make plans to attend all four days and you might want to even bring an extra person or two to cover those unavoidable conflicts. Send in your reservations early to assure a room and prepay to take advantage of the early payment discount. We have a 600 seat room, lets fill it!

See you in Chicago and if you have any problems, questions, or suggestions, please don't hesitate to call or write.

Stuart M. Katz
RSTS SIG Symposium Coordinator

(312) 298-7766
1011 E. Touhy Suite 360
Des Plaines, IL 60018

COMMERCIAL COMPUTER SERVICES, INC.
1011 E. TOUHY • SUITE 360 • DES PLAINES, ILLINOIS 60018 • 312/298-7766

February 21, 1978

SCHEDULE FOR 1978 SPRING DECUS SYMPOSIUM

TUESDAY APRIL 25th

Rosemont C & D Room	9:00 - 9:30	Welcome to DECUS for RSTS Users
Rosmenot C & D Room	9:30 - 10:00	RSTS SIG Meeting
Rosemont C & D Room	10:15 - 11:15	VAX for RSTS
Rosemont C & D Room	11:15 - 11:45	DECNET for RSTS
Rosemont C & D Room	2:00 - 3:00	RSTS version 6C
Rosemont C & D Room	3:00 - 4:00	RMS Optimization I
Rosemont C & D Room	4:15 - 5:15	RSTS Commercial SUB-SIG
Rosemont C & D Room	5:15 - 6:15	BASIC PLUS to BASIC PLUS II Conversion

WEDNESDAY APRIL 26th

Rosemont C & D Room	8:30 - 9:30	RSTS version 7A and the Future
SAS Room	10:15 - 11:45	Disk structure and optimization
Rosemont C Room	2:00 - 4:00	RSTS/E Disk papers
Rosemont D Room	2:00 - 4:00	RSTS/E Application papers
KLM Room	3:00 - 4:00	Digital Software Support and Field Test Procedures
Rosemont C Room	4:15 - 5:15	RSTS/E Utilities paper
Rosemont C Room	5:15 - 6:15	RSTS/E Internals Tutorial
Rosemont D Room	4:15 - 6:15	RSTS/E Application papers

THURSDAY APRIL 27th

Rosemont C & D Room	8:30 - 9:30	BASIC PLUS II optimization tutorial
Rosemont C & D Room	9:30 - 10:30	RSTS/E Open Discussion I
KLM Room	8:30 - 10:30	MACRO for RSTS/E
Rosemont C & D Room	2:00 - 4:00	RSTS/E papers
Rosemont C & D Room	4:15 - 5:15	To Be Announced
Rosemont C & D Room	5:15 - 6:15	Commercial Product Panel
KLM Room	8:00 - 11:00	RSTS/E and Commercial Systems

FRIDAY APRIL 28th

Rosemont C & D Room	8:30 - 9:30	BASIC PLUS optimization
Rosemont C & D Room	9:30 - 11:45	RSTS Wish List: Feedback Session
Rosemont C & D Room	2:00 - 3:00	Configuring and SYSGENNING
Rosemont C & D Room	3:00 - 4:00	The Anatomy of a Run Time System
Northwest/Transworld	2:30 - 4:00	DECNET for RSTS
Rosemont C & D Room	4:15 - 6:15	RSTS/E Open Discussion II

DESCRIPTION	CONTACT
1.) Subroutines to phonetically code last names/for storage and retrieval via standard phonetic techniques. 2.) Screen generator programs for the Tektronix 4023 buffered CRT terminal. 3.) Complete letter writing system using paragraph/letter files to build custom upper/lower case letters from form paragraphs. System allows insertions as well as a user composed paragraph.	Murray Stinson Scientific Placement, Inc. 2700 South Post Oak, #2430 Houston, Texas 77056 713-621-9811 4/1/77
1.) DATA BOSS/2 - Data Base Management Systems and Applications Monitor. Conversational setup, an English-like Inquiry Language, fast retrieval of millions of records via up to 65 keys. DBMS provides device independence and enables I/O with any ASCII file or device.	John H. Wright Florida Computer Inc. 99 N. W. 183 Street North Miami, Florida 33169 305-652-1710 4/1/77
"FIPS" - Financial Information Processing System. An easy to use computerized accounting system designed by accountants for accountants. Includes general ledger, payroll, payables, cost accounting, financial statements, etc. Features: ● Automatic Interface to General Ledger ● Complete Audit Trail ● Unique Report Formatter ● Company Consolidation and Overhead Budget Analysis	Tim Schroeder or Mike Niernberg Environmental Developers, Inc. 13693 East Iliff Avenue Denver, Colorado 80232 303-758-4619 4/1/77
The Statistical Programs for the Social Sciences/Carleton (SPSS/C) for the PDP-11. It is written in FORTRAN IV and tested under DOS V9, RT-11, and RSTS.	Graham Kimble or Carl Henry Computer Center, Carleton College Northfield, Minnesota 55057 507-645-4431 Ext. 204, 504 4/1/77
RSTS/E OMSI SORT-1 - Fast SORT for Large Files. Machine language SORT Package compatible with DEC BASIC-Plus SORT runs up to 50 times faster than DEC version. Sorts files containing over one million records. Very attractive price with OEM and multiple CPU discounts. Runs on version 6B as normal 16k word user job without software or hardware modifications. Chain out and back or run stand-alone.	OMSI Computing 4015 S. W. Canyon Road Portland, Oregon 97221 503-248-5900 4/1/77
PDP-11 TO /370 2-WAY DATA TRANSFER - Two programs: 1.) RSTS/E V6-B Basic-Plus program to transmit/receive data files to/from IBM/370-158, using BINARY I/O, on a full-duplex line. 2.) IBM VS1 ASSEMBLER program, using BTAM, may be called by COBOL or other /370 languages, to send, receive, rename, or kill files on any PDP-11, connected by dedicated full-duplex line. This system is currently in use in our installation to transmit data files in both directions between our /370-158 and a PDP-11/34 which is 200 miles away, connected by a dedicated line, as well as to our PDP-11/70, which is in the same room, and is connected by another dedicated line. Current transmission rate is 1200 Baud.	I. F. (Bud) Dawson MacMillan Bloedel Limited 1075 West Georgia Street Vancouver, B.C. Canada V6E 3R9 7/1/77

DESCRIPTION	CONTACT
HASPBOX is a communications front-end for RSTS/E. It allows full-duplex data transmission between RSTS and IBM/360/370 mainframes via HASP protocall. The system allows up to 4 input and 4 output streams to be operating concurrently, and also supports a HASP operator's console. The system has hooks into the standard RSTS batch processor and line printer spooler combined with a input Queue system. RSTS-LINK is a communication link for use between one or more RSTS based PDP-11 systems. The link provides a fast, low-cost link between the systems allowing transfer rates of up to 75KB. The link is fast enough for use in a real-time application such as data-base inquiry. The system has hooks into the standard RSTS batch processor and line printer spooler combined with a input Queue system. This link creates a true remote-batch facility for each interconnected RSTS system.	Gerry Moersdorf Software Results Corp. 455 West Seventh Ave. Columbus, Ohio 43201 614-421-2094 10/1/77
BMDP Statistical Programs for the PDP-11. A complete implementation of the February 1976 version of the programs. Twenty-six different programs. Entirely in Fortran and Macro. Developed under RSTS, installation under RSX, LAS, and UNIX is also possible. Distribution includes 2400' tape with complete source code, test data, sample output, and full documentation.	Dr. James Krupp Software Development Inc. Middlebury College P. O. Box 500 Middlebury, Vermont 05753 802-388-7976 10/1/77
1.) TAM, a format-language-driven terminal-handling front-end software module for transaction-processing applications under RSTS/E. 2.) KDSS, a key-to-disk data entry subsystem under RSTS/E. (In both TAM and KDSS, the performance-critical portions are implemented in PDP-11 assembly language and installed under RSTS/E as run-time systems.) 3.) FSORT3, an ultra-fast run-time system capable of sorting up to 16 million blocked records or keys. Record and key formats are compatible with the RSTS-11 Sort package. The FSORT3 package contains modules for replacing SQWIK, MQWIK, and XQWIK. FSORT3 may also be used stand-alone. The package uses about 2% of the CPU time of its BASIC-PLUS counterpart and runs up to 30 times as fast in stand-alone wall time measurements. 4.) INTCOM, a set of BASIC-PLUS programs allowing two RSTS/E systems to communicate, including ASCII file transfer, via telephone line up to 300 baud.	Tom Evans Evans Griffiths and Hart, Inc. 55 Waltham Street Lexington, Massachusetts 02173 10/1/77
RMS-11/BASIC-PLUS 2 Integrated Financial Accounting System designed for Universities, Municipal Government, Government Agencies, and Commercial Users: Modules include: - Purchase Order Module - Accounts Payable Module - General Ledger Module - Budget Module - Vendor Module - Balancing and Audit Module	Dennis Sakamoto or Richard Rock SRB Computer Systems Ltd. 50 Crown St., Suite 175, Saint John, N.B. E2L 2X6, Canada 506-693-0620 2/1/78

DESCRIPTION	CONTACT
WORD-11: Word Processing for RSTS/E. Extremely powerful multi--terminal word processing system. Uses standard VT52 terminals and any printing device, including letter quality daisy wheel printers. Implemented as a runtime system in MACRO-11. Functionally similar to DEC's PDP-8 based word processing system.	Rich Marino or Mike Mayfield Data Processing Design, Inc. 6980 Aragon, Suite 8 Buena Park, CA 90620 714-994-4971 2/1/78
SAVER: Replacement for BACKUP for RSTS/E. Written in MACRO-11, provides fast, reliable disk to tape, and disk to disk backup. Writes PIP-readable files, maintains all file and accounting information; restore operation utilizes optimal UFD/file placement algorithms. Fast, efficient, with user-oriented interface.	Rich Marino or Mike Mayfield Data Processing Design, Inc. 6980 Aragon, Suite 8 Buena Park, CA 90620 714-994-4971 2/1/78
PAGE is a full screen video terminal text editing program designed to provide the user the convenience of cursor controlled text editing. Rather than entering "blind" commands to search, position, and modify text, PAGE displays the text in context on the terminal screen. The user can visually locate text to be modified, then position the cursor under the desired character or line and key in the change. The change is then immediately reflected on the screen. Both programmers and non programmers can easily learn PAGE. PAGE is ideal for producing and editing documentation.	Pam Biederman Interactive Information Systems P.O. Box 37403 Cincinnati, Ohio 45222 513-761-0132 2/1/78
IIS Program Dispatcher: The dispatcher is a facility to allow end users of RSTS/E to execute certain sets of predetermined programs without having to know or understand anything about the RSTS/E monitor or file structure. The user simply enters a "transcation code" which the Dispatcher validates for that user and maps to a program name. The Dispatcher can greatly simplify the task of teaching end users to use the computer, and requires almost no additional system overhead. The Dispatcher can also record system utilization information and can automatically log users back in after a system crash.	Pam Biederman Interactive Information Systems P.O. Box 37403 Cincinnati, Ohio 45222 513-761-0132 2/1/78
IIS Sort/Merge Package: The IIS Sort/Merge Package is a set of five programs designed to perform the functions of sorting and/or merging files. Files may be either record I/O or formatted ASCII and may contain the standard field types (character string, packed integer and floating point or ASCII). Up to ten fields in sixteen files may be sorted in ascending or descending sequence. It is also possible to chain to and from application programs. The critical sorting code is written in MACRO-11 Assembler language which results in an approximate 100-1 improvement in execution efficiency.	Pam Biederman Interactive Information Systems P.O. Box 37403 Cincinnati, Ohio 45222 513-761-0132 2/1/78
IIS Data Base Management System: The IIS Data Base Manager provides users with a comprehensive and powerful data management tool at an affordable cost. The system incorporates many of the facilities found only in larger data base systems at a fraction of the cost, and is designed to be very easy to use. The system provides data base access and update capability simultaneously to a large numbers of on-line users. Each user may access up to 128 files at any one time.	Pam Biederman Interactive Information Systems P.O. Box 37403 Cincinnati, Ohio 45222 513-761-0132 2/1/78

RSTS MARKETPLACE

DESCRIPTION	CONTACT
IIS Data Logger: The Data Logger is a data management facility for sequentially organized information. The Data Logger can store and retrieve an almost unlimited number of files on either disk or magnetic tape, and is provided with two application systems: a Data Recovery system which allows users to "log" on-line updates to critical files and provides for the recovery of these files in the event of a "hard" system crash, and the Transaction Accounting system which records user logins and logouts and can record the execution of individual programs. Facilities are provided for recording user generated information and extracting and printing billing information.	Pam Biederman Interactive Information Systems P.O. Box 37403 Cincinnati, Ohio 45222 513-761-0132 2/1/78
MIRCOM: A software package wtitten in BASIC/PLUS to transmit messages or files of any type between two or more computers operating under RSTS/E. - No special communications hardware is needed. The system requires only a single standard serial interface operating at any standard speed on each machine. - A message send/receive protocol is incorporated which allows binary data to be transmitted and received and provides recovery procedures for data transmission errors.	Rex Hill or Gil Jost Diagnostic Computer Division Mallinckrodt Institute of Radiology 510 S. Kingshighway St. Louis, Missouri 63110 314-454-2181 2/1/78
INTEGRATED ACCOUNTING APPLICATIONS: From one of the pioneers in commercial data processing using RSTS. Completely integrated core accounting applications. Ready for immediate delivery. Completely interactive. Extensively documented. Easily installed. Fully supported. Written in BASIC-PLUS. Ideal for OEM's, service bureaus or end users. Includes: ● Accounts Payable; ● General Ledger; ● Financial Reporting; ● Accounts Receivable; ● Payroll. Features: ● Complete integration; ● Multi-entity consolidations; ● Extensive audit trail; ● High degree of reporting flexibility; ● CRT or hardcopy input.	Bob Sharp PLYCOM Services, Inc. P. O. Box 160 Plymouth, Indiana 46563 219-935-5121 3/1/78

UNIVERSITY OF PRINCE EDWARD ISLAND
CHARLOTTETOWN
PRINCE EDWARD ISLAND

COMPUTER CENTER

March 10, 1978

Mr. Murray Stenson
RSTS Newsletter Editor
Scientific Placement Inc.
2700 South Post Oak Rd
Suite 2430
Houston, Texas, 77056
U. S. A.

Dear Mr. Stenson:

Could you publish the following in your next newsletter?

I would like to hear from anyone who has:

1. A version of RSTS Batch that supports indirect command files. Nesting is not required.

2. A fast, single-pass linker for RSTS/E FORTRAN programs suitable for use by beginning students.

My address is:

Computer Centre
University of Prince Edward Island
Charlottetown, PEI. C1A 4P3
CANADA

Thank you in advance.

Sincerely,

J. W. Hancock
Director - Computer Centre

Grand Canyon College

3300 WEST CAMELBACK ROAD • (602) 249-3300 • POST OFFICE BOX 11097 • PHOENIX, ARIZONA 85061

Director of Computer Services

March 14, 1978

Mr. Murray Stinson
Scientific Placement, Inc.
2700 S. Post Oak Road, #2430
Houston, TX 77056

Dear Mr. Stinson,

If possible, please place this letter in the next issue of the RSTS newsletter.

At the Fall 1977 DECUS Conference, a small group of COBOL-11 users met together to talk about common COBOL problems, needs, wish list items, etc. We would like to expand this group by contacting as many COBOL-11 users as possible. Our immediate goal is to give DEC some guidance on what we as users would like to see in future releases of COBOL. Eventually, we would like to form a COBOL Special Interest Group.

If you would like to get on our mailing list and participate in the initial "wishlist" survey, please send me your name and address or see me at Chicago DECUS. Our goal is to have the survey results compiled and ready for discussion with DEC at the Fall DECUS Symposia in San Francisco.

Sincerely,

Frank Anderson
Director of Computer Services
Grand Canyon College
3300 West Camelback
Phoenix, AZ 85017

FA:dch

Ron Webster
University of Toledo
Computer Services
Toledo, Ohio 43606

Dear RSTS User:

The University of Toledo is attempting to develop a charge-back system for our PDP 11/40 and 11/70 RSTS timesharing systems. Currently we support only academic users but will soon add administrative applications. We would like to get some input from other installations before making any decisions on rate structures. If your installation actually charges for services or just keeps internal records, we would like to hear from you. We would be particularly interested in what resources are monitored and what the associated charges are. All information will be kept in strict confidence! We would greatly appreciate any help you may give us.

If I receive enough responses to draw any general statistics from, the results will be published in the RSTS SIG Newsletter. This I feel would be quite interesting and particularly important for someone who is attempting to develop a charge-back system.

Need your help.

Ron Webster

RSTS/E Questionnaire for 1977 Fall DECUS Symposium -- San Diego, CA; Nov-Dec 1977

A. How do you rate this meeting ?

!! Very informative and helpful !! Good !! GOOD !! good. !! Very Informative !! Excellent !! 2.
!! Very Helpful !! Good !! Very Good !! Good-

A1. List the sessions that were the most useful to you. Why ?

!! small user !! RSTS FEEDBACK, Direct communication line between RSTS developing group and users
!! performance Evaluation - we have a fairly stable system re; hardware and software. This is our current big bugaboo. Mostly the panel helped me. Feedback - obvious only !! Performance criteria seminar; Feedback session; Both solicited specific problems and stressed shared information. !! Improving RSTS Usability; Concepts for Systems Operations
!! DEC - User or User - User Interactive sessions; RSTS Performance Evaluation; The tutorials were generally helpful
!! Program standards - nice meaningful presentation without being too verbose; RSTS Usability - interesting and fun; Systems Access - very meaningful !! Tutorial & Birds of a feather & Hospitality Room. !! Wishlist & Feedback; Machine Room informal sessions with other users; Device drivers !! Discussions relating to use of RSTS; Coming Improvements; Tips on Use; Discussions of other users' problems and solutions !! The unstructured sessions with considerable interaction from users, because there was no specific topic. !! New User Session - This was well done, being broke up into small groups of specific topics. CUSPs workshop - informative hearing from other users. !! New Users' mini-sessions - smaller, the 'talkers' weren't expounding; chance to get really specific. EDUSIG meetings. !! RSTS New Users' session; informative on a wide range of subjects; a good format for interchange of experiences and knowledge (don't forget the new and/or small RSTS user). BASIC-PLUS-2 Language Tutorial - allowed me to evaluate Pros & Cons of B+-2; URECA birds-of-a-feather; my first exposure to URECA !! BASIC-PLUS-2, RMS, all interactive sessions (feedback workshop); New Users
!! Improving RSTS usability; System Access Mangement; CUSP workshops !! CUSPs Workshops !! 1. 6C sessions - advance planning; 2. Wishlist - get in 2 cents' worth; 3. CUSP workshop - get in 2 cents' worth !! Sessions with DEC; BASIC-PLUS versus BASIC-PLUS-2 !! New Users' Session - because many subjects were discussed in small groups. I could select the subject. One problem - I wanted to take part in other mini-sessions, but time ran out. !! CUSP sessions; info on 6C
!! RSTS Introduction and other session; Educational system security - etc. RSTS New Users' Session !!! !! BASIC-PLUS-2 sessions, RSTS 6C sessions; CUSP workshops !! New Users' Group - Open discussion; papers; generates new ideas
!! 'Major ingredients of a GeneralAccounting Package', 'On-Line A/R Entry' and 'RSTS/E New Users' Session' were all useful because they showed me some useful techniques and answered some specific questions which I had. !! RSTS Device Drivers - I'm going to write a file structure for floppies; RSTS hospitality suite - a chance to ask some detailed questions.
!! RSTS Wishlist discussion - gives shortcomings of RSTS; RSTS New Users' Session - can relate similar problems; RSTS Feedback - get solutions to common problems not in Newsletter or not out in Newsletter; RSTS Device Drivers - fills needed documentation on how to implement one; DYNPRI paper gave good insights as to improving throughput of a system. !! Device Drivers; I need to write a driver; MACRO; Table Software - nice techniques; DECUS Informal - learned a lot; ZAP - need security; Education Papers - my area !! RSTS Performance Evaluation; DYNPRI; RSTS Wishlist; Timesharing standards; Basic-Plus documentation; and any session that was structured or directed

A2. List any sessions that did not live to your expectations. Why ?

!! RSTS Small Systems - the discussion was deversified and didn't get much action. !! New Users' Session !! The posted RSTS papers were generally disappointing. Some presentations lacked depth or organization. Since library automation is neither

a commercial nor an educational application, many of the topics were not of interest. I think an effort should be made to present papers for the fringe RSTS applications, as well. || I thought all the RSTS sessions were well done. I didn't like the VAX presentations; they smacked too much of a sales job. || Those Cancelled || None || None || "RT-11 to RSTS Conversion": should have been entitled "CTS300 to CTS500 Conversion". It did not occur to me from either the title or the description printed in the Symposium guide that this session would be about DIBOL exclusively. (This session was presented by DEBUG, not RSTS.) || MOVE NEW USERS TO FIRST DAY ! || In general, formal paper presentations by users were not very helpful. || A lot of the sessions were geared to large systems and technical heavy's. I'm not one of "them" animals. || I could have done without the salesman-oriented paper presentations (Filling, BASIC + Documentation, Slash/Design). || ? Hard to say - || Many of the papers were too general - possibly due to time constraints. || MACRO under RSTS -two different sessions, both too general. Non B+ RTS - didn't show up. Copy Center ! machine never came up. || DEC cancelled RSTS Fortran conversion paper, Non-B+ RTS; RT11 to RSTS Conversion was for DIBOL & CTS500, not Fortran. || CUSP's: not what I expected; expected code to do different things. || Any that the speaker read each slide and offered practically no added information.

A3. What sessions need to be added at the next DECUS Symposium ?

|| MORE Application sessions, perhaps have SYSGEN tutorial || Inexperienced users' session; New small commercial users' session || More on evaluation, ie evaluation of disk throughput, evaluation and comparison of languages, ie benchmarks. || Expanded New Users's Session - Advertise in Newsletter so that more new users will come the Symposium. || How about a high-level techniques Jam session ? || Earlier, pre-DECUS sessions, particularly for new users; Sessions on "unpublished" RSTS features, switches, etc. || More for new users - operations, using CUSPS, using RSTS, optimizing system, maximizing throughput; As Educational User - need a forum for exchanging info, programs. || I would like to see general sessions with no specific topics but with RSTS people from DEC available to answer any questions from the floor. This would include technical and management types from DEC. || I can't imagine - the 4 days are too full now. || More on system-monitoring (performance, accounting, etc). || Would like to see people with good, shareable, programs make them available at swap and give sessions on how to install/use them. || Improving system performance; optimizing BASIC programs || A Glossary session for new users, EARLY; During my new user session, I discovered that few (if any) understood what a Run Time System is. || More small/new user sessions of a workshop nature... 1. how to tune machine (11/34's); 2. CUSPS; 3. measure disk performance. || More interaction on techniq ue sharing; CUSP practices || More small discussion groups - (new users' group); general technical questions ||General comment: Many of the presenters of 30-minute papers did not finish their talks properly. Perhaps 45 minutes should be the minimum for a session. || Advanced users' session, along lines of new users' session. || Hey ! A complaint about scheduling: I come for 1) technical ideas/interchange (DYNPRI considerations, MACRO under RSTS, RSX SIG's FCPIACP tutorial...), then for 2) interaction with DEC (wishlist, informal discussion). BUT ! the technical sessions all pile up and conflict on the 3rd and 4th days! Can't we have the SIG sessions and DEC product panels spread over all 4 days ?

A4. Were the handouts for the sessions helpful ? How can they be improved ?

|| Yes. || Yes, helpful. || Yes || Somewhat. The handouts should both summarize the session and expand on the sessions. Some handouts had nothing to do with the sessions. || The handouts were helpfuld. || Tremendous. || Yes - not enough; should have been more. || Yes. Any chance of getting them a week or so before where I might have chance to read before coming ? || Yes. || Handouts should contain where to find more information, who to contact. || Yes. || Handouts were very helpful. || Very helpful. Improvement would be made if the speakers wouldn't read them - we can do that on our own. || Excellent; possibly a standardized format at least for basic information regarding title, author, author's firm or association, locality, etc. || Yes. || Tremendously. Make it more complete. Maybe include the latest newsletter(s) for the benefit of people not on the mailing list. || The Preprint is Wonderful. Finest Kind ! || VERY Helpful. || Yes, some were too long. || Yes, more complete; handouts not specifically related to a presentation would also be OK. || Yes - more. || Look good. || Yes - more detail. || They are useful because, if I have to miss a session, I can choose to miss one which has a good write-up. Alternately, I can come prepared to ask questions based on the written material. || Yes- need more. || Yes, but need more technical info. a la Anton Chernoff's paper on Device Drivers for RSTS. ||More handouts or all put into Proceedings or SIG's. || Yes - But there should have been more in place of (or as well as slides) as the slides were unreadable - why not place projectors on both sides of the large rooms, halfway

down each wall ?

B1. What new projects and what new working groups would you like to see started by the SIG ?

!! A working group on available STATISTICAL packages in order to document not only their availability but also their scope and efficiency. !! Performance Evaluation SIG !! RSTS Performance SIG !! BASIC-PLUS-2 - any way of list SIG members & machines & operating systems. !! Product dissemination-information (via newsletter, etc.) to users concerning DEC (and others!) products available for RSTS, pricing changes in hardware & software, etc. Working group to "welcome" new users. !! I feel the SIG is doing a good job and should continue working on current projects. Strengthen current organization. !! This area seems to be quite comprehensive. !! How about a small user subsig ? !! A library committee to review the DECUS library programs which are supposed to run on RSTS. !! Performance Evaluation, and Upgrading !! Fast Response Information. I would make available my system after 6PM EST for those who would call at their expense to get news. I would like to see SPR's and Newsletter articles included. !! Have we dropped standards and documentation ? !! New User "Help" Group !! I would like to encourage the (quick) development of a strong recommendation about system. !! Resource management/optimisation. !! In-House Computing. !! Perhaps one to do with inter-machine communication, especially with non-DEC machines. !! MACRO SIG.

B2. What changes would you like to see in the newsletter ?

!! I haven't read the newsletter yet. !! Better printing. !! Haven't received the newsletter yet. !! Get DEC to clear it more quickly and get it out in more timely fashion. Continue to solicit articles from members. !! I would like to see it produced more regularly. !! I haven't seen enough of them to tell. !! Slightly more editorial descression. See contain too many low or non-content letters. !! Great, I like it ! Can we get more issues ? !! None. !! I guess the problem is that user participation is low - what I'd like to see is more exchange of technical info.

B3. How do you feel the DECUS library could be improved for you ?

!! Update programs; newer, more useful, programs (you can only use so many games); RK06 support. !! Haven't used the library yet. !! Bring programs up to STANDARDS. !! More frequent notes on additions or updates. Pursing outdated material. Solicit updated programs for "old" entries. !! By simply notifying members more often of new submittals. !! I'm happy as is-- !! If DECUS is to have a centralized useable library, it needs access to a machine(s) capable of distributing programs on all types of media. Library programs should be reviewed by a committee se up within the applicable SIG. The committee should remove from the library programs which will not run properly. If the author of a program cannot or will not modify the program, the committee should decide the fate of the program as to making it run or removing it from the library. !! Move programs from DECtape to mastape ! !! Make library part of DECUS booth system so we could collect and pay for programs here. !! Quicker notice of new additions. !! Set standards and urge users to meet them. !! Faster response; better quality listings/output. All documentation in RUNOFF format. !! Need to get a catalog. !! I haven't used it yet so I don't know. !! I know this is just more work - but it's frustrating to know that some nifty program is in the DECUS library but you don't have a new enough catalog to show it. Can catalog updates be available on some regular basis ? !! Put on RK06.

C. What kind of RSTS user are you:

1. How many years have you used RSTS ?
2. How is your machine used ?
3. What is your job function on the machine ?

4. How many DECUS Symposia have you attended previously ?

3; scientific, statistical; system manager; 2 ;;3 months; updating data base & management report; manager+programmer+operator; none ;; 2; educational & research; administration, research; 3 ;; 1; educational; system manager; 0 ;; 9 mos; commercial - order entry etc; programmer, DP mgr (everything); 0 ;; 1; RJE station, text editing of source code, data entry; 25% system manager, 75% programmer & user; 0 ;; 1; commercial timesharing; software specialist; 1 ;; 1; educational; programming support; 0 ;; 1; commercial - production calculations; ??; 1 ;; 4; instructinal computing; manager; 1 ;; 1; education/administration; director of computer systems; 1 ;; 3.5; program development for commercial users; manager of programming; 1 ;; new; not up & running yet; system management; 0 ;; 1.5; education; system manager; 0 ;; .5; estimating, engineering, accounting; system manager, programmer, back-up operator; 1 ;; .33; (proposed) muni govt accounting, utility billing, police services; 0 ;; 1.5; data entry, retrieval, summary reporting; system manager; 0 ;; 1.5; data processing - payroll accounting, student records; systems analyst; 1 ;; 5; commercial & engineering; manager; 3 ;; 4; commercial; data prostitute (all functions); 4 ;; 1; commercial; manager; 0 ;; 1; education; director;computer services; 0 ;; 1; education only; manager-teacher; 0 ;;1; commercial; ??; 3 ;; 3 mos; order entry, RJE to 370; sysgen, optimisation, software support & modification; 0 ;; .5; time-sharing/data entry; time sharing coordinator; 0 ;; <1; one for online library circ., one for data entry & RJE to 370/158; software group mgr; 0 ;; 1; program devlopment, text processing, math/scientific, industrial control; sytem manager; 2 ;; 3 mos; engineering applications, hope to get RT11 users to try RSTS/E; system manager; 1 ;; 6 mos; educational (teaching & administration); everything; 0
;; 2; Educational; Timesharing Manager; 0

San Diego Feedback Panel
Extracts

For those of you who haven't yet had an opportunity to attend a DECUS Symposium, the "feedback session" is a mechanism for two-way dialogue between us and DIGITAL. It's proven very popular at the past couple of Symposia, and works like this: there's a large box in the lobby area, labelled "Channel 13", into which we drop questions, comments, brickbats, and/or kudos for DIGITAL. For the first couple of days the collection process goes on, and we sort the received material by general area. Each item is given to one or another DIGITAL representative - DIGITAL is committed to respond to each item presented to them. Typically, the panel consists of people from Central Engineering (the RSTS developers), Software Service, and commercial product planning.

What follow are some extracts (what I was able to transcribe) from the Feedback panel in San Diego. Because the transcriptions are mine, DIGITAL is not responsible for this material.

Respectfully submitted,

Bill Sconce
Channel 13 (Wishlist) Editor

Question. DEC has indicated little or no more work will be done on Basic-Plus due to: (1) hitting 16K boundary; and (2) Basic-Plus-2 is getting funding. Will this soon happen to RSTS because: (1) Monitor is already very large; and (2) VAX is getting funding ?

Response. This Basic-Plus / Basic-Plus-2 issue will be re-evaluated. The main emphasis will possibly be to provide much more Basic-Plus functionality with Basic-Plus-2, as well as addressing the compile and task-build problems, and generally improving the total throughput of Basic-Plus-2. For the larger question, Digital has no intention of dropping suport for RSTS/E in favor of VAX.

A User's Comment on the Basic-Plus'es. Basic-Plus needs a modular environment whereby certain options can be appended or deleted as needed by a program. Basic-Plus-2 is NOT acceptable to a commercial user, because of the inordinate amount of time in the taskbuilder. It appears that Digital took a step backward to a batch environment from a rapidly-accessible system...

Question. Basic-Plus-2 on all systems (RSX, VAX) should have the FIELD statement. This concept is basic to the design of many existing Basic-Plus programs, and no easily convertible concept exists in Basic-Plus-2 for transport to other machines.

Response. Because of user requests (in RSTS/VAX meeting, for example), a serious look will be taken at this, and to re-examine the Basic-Plus-2 language standard. The technical issues are not serious.

Question. Improve communications (internally) between DEC hardware and software personnel.

Response. Hardware people are being trained in software, and some software people are being trained in hardware. They do talk to each other. Autopatching is one example. Certainly error logging is a classic case - it was conceived by a hardware person, initially. VAX was designed with the problems of joint maintainability in mind.

Question. Increasing number of manuals, frequent re-issues, and high prices are quite a burden on student-type users, and on the University bookstore.

Response. For educational institutions, the product lines can make arrangements concerning reproduction of manuals for use on campus.

Question. A course on RSTS/E internals.

Response. In the past (on several occasions), Digital offered to provide a seminar which was essentially the course given to software specialists, with marketing issues deleted. The response was underwhelming. Typically, there would be about four to five inquiries and one tentative enrollment. Then, the one person who did want to enroll wouldn't meet the prerequisites. (Prerequisites would include the equivalent of the entire 6 weeks' instruction in RSTS/E, plus a substantial amount of hands-on experience with the system, plus a thorough understanding of assembly-language programming on the PDP-11.) This sort of request has been coming in for a number of years; every now and then it's tried again, and again doesn't make it. If there is in one localized area a group of users who want to do it, their sales representative can contact Educational Services, and it'll be tried again.

Question. Whatever happened to the RSTS/E internals manual for users that was supposed to be available someday?

Question. How I wish I had an internals manual...

Question. The H.A.L. Corporation has program logic manuals for their system software, even though they frequently change their software. You don't have to sign a waiver before you read them. What makes Digital's circumstances so different that they cannot do the same ?

Response. As many users undoubtedly know, there exists an under-the-counter, very preliminary document purporting to describe the internals of RSTS. If Digital were ever to publish such a document, one problem would be that there would be an obligation for it to be correct. It would be a disservice to distribute misleading material. (That document CAN be misleading; people trying to use it have had some surprises.)

The real source of that kind of information is available, although at considerable expense - that's the listings. Trying to take the listings and derive from

them a manual is a considerable effort - an effort Digital has often thought they'd like to make. However, given relatively fixed resources, if they did that, they'd have to not do something else. The decision has always been to do the "something else". The question keeps coming up, but the decision keeps being made the same way. Finally, this issue arises: would people be willing to pay a substantial sum of money to have such a document ? It could be questionable to ask the entire community to pay for production of a manual which could be used by a relatively small portion of it.

(Editor's Note.) It's interesting that RSTS' technical history has been the reverse of the "classical" one. Many RSTS users have gotten started with minimal knowledge of internals; even today, few of us do much MACRO coding. Most of us have worked on applications first, and only later come to be interested in guru-ism. In the "classical" case (and I learned this way, on another manufacturer's system), applications came last. The FIRST thing was mastery of several nitty-gritty type manuals, the use of core dumps, and a really awesome book of encrypted error messages.

(Obsolete for a little while, this education may again be useful: now that RSTS/E users can write RSX-style, it's possible to get core dumps!)

Question. There were several questions concerning Fortran - whether Fortran-IV-Plus would be offered, RT Fortran versus RSX Fortran, etc.

Discussion. Fortran-IV-Plus: Digital talked of circulating a questionnaire soliciting interest in a possible product for RSTS. It now seems that the questionnaire will not be necessary, as a Fortran-IV-Plus offering for RSTS/E is in the works. It will be an RSX flavor Fortran.

It is also likely that the RSX-flavor Fortran IV will be made available.

Question. Fortran run-time diagnostics are very obscure! "? - M trap to 4 at user PC xxxxxx" doesn't say much. Can we have more informative diagnostics ?

Response. There isn't much enthusiasm for documenting RT-11 type messages for RSTS/E, but they may have a look at getting Fortran itself to do a better job. Incidentally, the particular message asked about is from the RT-11 emulator, and would indicate that Fortran lost control. Subroutines in MACRO could be responsible, too.

Since MACRO was brought up, it was also pointed out that someone coding in MACRO can do something bad to himself, but unless privileged, cannot do anything bad to the system or to other users. MACRO is not "subversive" in the sense of getting around any of the normal facets of system security.

Question. ADVENTURE as a supported CUSP.
Discussion. (Laughter).

Question. Move the MFD to [0,2] from [1,1] so that it may not be accessed, and slowed down by users or Basic-Plus-2 files, or anything else.

Response. The answer is not to move the MFD, but to move out the files causing the crowding. Work is being done on how to do that. When a solution is arrived at, it will be published in the Software Dispatch or included in the next release.

Question. Are there plans to permit more than 1700 PPN's on the RSTS/E system ? I'm in an educational institution with some 3000 students per semester, and this limit causes innumerable problems.

Discussion. No, because of file structures in RSTS/E, and because of large files, it won't be possible to lift that limitation. Lifting it would require a drastically different file structure, and there are no plans for doing that.

Question. A facility for creating contiguous output from the Task Builder is needed. This would significantly improve throughput on large, overlayed tasks. To a lesser extent, the same facility would be helpful for .BAC files, too.

Response. Beginning with version 6C, TKB will automatically try to build a contiguous file, and will fall back to non-contiguous if insufficient room exists. They don't see that much advantage for .BAC files, but one can do it with PIP, using /MODE:16.

Question. Utility to print out locations of clusters on disks. I am about to write my own utility to do this. I hate to waste my time this way.

Response. Such a utility has existed informally in the field for about two years. If it can be done in a reasonable length of time, the author (not in the RSTS/E development group) will upgrade it to work under 6C, and submit it either to the DECUS library or to the RSTS newsletter. (Applause.)

Question. Improve error copy to accurately log (all) errors within the system. IE, disk seek errors get logged individually, but Unibus timeout errors do not get logged.

Response. There seems to be some confusion as to what a Unibus timeout error really means. It can happen if you do a peek to an odd address, or if you are writing a MACRO program and somehow use bad code. That could be equivalent to "?M - Trap to 4". It happens by the USER doing something wrong (as opposed to a hardware problem). Although the error description mentions the Unibus, that's misleading. There's nothing wrong with the Unibus.

Question. Performance measurement and optimization is important enough to be a whole new RSTS/E manual. It is also important enough for there to be a fully

documented and supported measuremend package. It is even important enough in my my opinion that one should be able to optionally generate a monitor with hooks in it for Job-by-Job (and file-by-file?) I/O statistics.

Discussion. This one may want to be brought up again, as the question wasn't fully resolved.They were pretty clear, however, that they don't want to tell us how to interpret STATS until they know what the numbers mean.

Question. Will DEC personnel ever discover some other word than 'functionality' to describe anything nitty-gritty ?

Response. 'Our consultants have informed us that there is no such word.'

Question. We have on occasion spent many hours debugging a program to find that an embedded comment in EXTEND mode was terminated by '& <LF>'. The line feed causes the next line to be treated as a comment instead of as a Basic-Plus statement. This cannot be detected from a print-out. Some way to disable line feeds in EXTEND mode (or after ampersands) is needed.

Response. An optional patch will be looked into.

Question. Multi-CPU RSTS which would let two CPU's simultaneously use an update-mode file.

Response. Presumed: that the question was about sharing a disk drive in dual-port mode. That's not supported, and there are no plans to support dual-port disks.

Question. Development of interactive and hard-copy graphics under RSTS/E.

Response. Graphics software is not directly supported under RSTS/E. VT11 support, for instance, would be difficult, especially considering RSTS' crunch for core. Only exception is (in 6C) a bunch of subroutines that support the VT55 terminal. (Editor's note: RS232 hard-copy plotters and the like can be hooked up, of course.)

Question. Please investigate and comment on possibilities of a more modular RSTS/E, with a defined baseline system configuration. It should be such that users could insure a basic functionality at a given small (6C-size) system.

Response. It is definitely realized that both modularity and size are problems requiring continuing attention. For instance, large-file support will guarantee that the next version of RSTS/E will be bigger in those installations which use it. For that reason, large files will be a modular addition, so that its overhead can be declined, if desired.

Growth admittedly compounds the problems with small users. Digital knows that this problem must be addressed, and although it hasn't been decided how it should be handled, we are assured that small users haven't been forgotten.

Finally, someone offered these observations.

For what it's worth... When we were shopping for a way to augment our timesharing capacity beyond our present 11/45, we almost went to a non-DEC vendor who offered us

1. Source-language distribution and updating of ALL software

2. Nice machine architecture, with some features similar to VAX

3. ability to emulate IBM HASP and CDC RJE terminals

4. ability to interconnect multiple machines with user transparency, ie the user can go to a terminal on any machine and access files located on another machine without having to know whose they are and without having to say anything special to move files.

In the end we stayed with DEC because the users were unwilling to have two dissimilar BASIC's (and to them the above goodies were less important than they are to us system types). But if the other vendor had offered something nearly as nice as Basic-Plus they would undoubtedly have gotten the order.

In the course of evaluating several competing systems we found that several machines of higher performance than the 11/70 can outperform RSTS up to a dozen or so Jobs, but bog down beyond that point. So RSTS/E is doing something right.

To: Roy Swatzell
From: Bill Sconce

Subject: Standards Discussion - a tentative draft, November 1977

Dear Roy:

As promised, here are some of my thoughts on the issue of program standards for Basic-Plus. The first thing to do is to thank you for volunteering to spearhead the SIG effort; this is an area which obviously interests many of us. Also, starting with something in print is no doubt the best approach. We don't all have to agree with the first drafts (and certainly we will not all agree with the old 6A DEC standard), but still, the dialogue must proceed by reference to something concrete. (Actually, I don't suppose that all of us will ever agree, period. Any useful standard will probably need a certain resilience.)

This is a collection of personal views; I offer them in the hope that they will stimulate discussion. I trust that no SIG member will be reticent about criticizing them, or about offering their own thoughts.

First, something about philosophy. Programming standards are always restrictive to some extent, and can make coding require more work. This is certainly the case for mandatory commenting, for instance. On the other hand, if they are properly conceived, standards can also be the source of considerable savings in time and mental energy. For example, the most popular part of those standards presently floating around is probably the line number allocations - and this is partly because the creative job of laying out the major structural blocks of a new program proceeds more smoothly if a prototype, or skeleton, can be relied upon. Of course, having many programs laid out in a parallel fashion will make all of them easier to maintain later on.

About restrictiveness, though - it seems to me that a comprehensive standard should have enough detail in it to cover most of the commonly-encountered coding situations. This means that standards should be written to be complete, so that a programmer wanting a standard reference for coding in a certain situation will have something to rely on. It does not mean, however, that we should forget about human judgment in the application of standards to each real-world program. The basic intent of any standard useful to the RSTS SIG will have to be reliance on basic good sense, and provision of enough options to be flexible.

I'll structure my comments around the DEC document which you published in the Resource Sharer. I guess it's obvious that the old DEC standard would have needed some updating to accommodate EXTEND'ed format, but I'll just mix those items in with the rest.

1.0 ORGANIZATION

My biggest point concerning program organization is the need for facilities to handle libraries of installation-wide and system-wide modules (routines and functions). Installation-wide modules are those subroutines and functions which should be available to any program written to run in a given shop. By 'system-wide' I mean a group of related programs, probably developed in parallel - 'Payroll' might be such a system in some shops, for instance, and could need modules such as a file-access routine or a particular date-printing function. Thus, a given program might need to include code from three sources: the installation library (global library), the 'system' library (regional library), and unique code written for the specific program (local code).

Because the only way modules can conveniently be combined in Basic-Plus is via the APPEND facility, setting up libraries means allocating a hierarchy of line number allocations. The system manager has to keep a table of line numbers used for each routine or function to be made available to the user community at large; the project leader has to do the same for each project's 'regional' library. Although not every installation (or project) might choose to implement code libraries, if they are ever to be possible in any natural way, the coding standards must allow for their allocation.

As far as I can see, all library material should be in the form of either functions or subroutines, for modularity of program architecture. This would require three line-number blocks (global, regional, and local) in each of these program areas:

1. General Description Narrative {1.3.3.1}
2. Definition of Variables {1.3.3.3}
3. Function and Subroutine Descriptions {1.3.3.4}
4. Dimension Statements {1.3.4}
5. Initialization and loading of constants {1.3.6.1}
6. Function Definitions {1.3.6.4, 1.3.7}
7. Subroutines {1.3.6.4, 1.3.7}
8. Error Handling {1.3.6.5}

NOTE

A. {x.x.x} refers to section numbers in the DEC program standards document. B. It's possible that (beginning with 6B) one line number will suffice for any given module's requirement in a particular category, except for GOTO objects, etc. C. ON ERROR handling may not need line number allocations in all shops - more below.

On error handler organization! I've seen shops where all errors must be fielded in a single block of code, usually at line 19000. While this has the advantage of putting all error recovery where it can be easily found, it does have drawbacks. First, the program's logic is split in two: to understand what's going on (in a real-world program), you have to be able to combine main-thread code with the error code, and these are usually some distance apart in the listings. Second, the error code itself becomes nearly incomprehensible, since it ends up containing procedures for a hodgepodge of different processing situations. The latter problem is compounded by the fact that many times, error-handling code suffers from the poorest commenting. After all, comments here must not only explain what a statement does, but also make notes concerning what main-thread routine the error being fielded pertains to.

There is at least one alternative, especially where programs are modularized by use of subroutines and/or functions. Here, each subroutine or function capable of generating an expected error resets ON ERROR GOTO. On initial entry, it points the error trap to a line within itself, and then resets it to the standard line before exit. This gets around the problems above, and improves the 'stucturedness' of a program - but some caution and care are necessary as well, obviously. One possible corollary is that most lines capable of causing errors might be best coded into subroutines or functions, where this convention is being used.

A final point about organization and line numbering. With version 6B, there's no longer any need to start a new line just because you've used up 256 characters. Obviously, any object of a GOSUB or GOTO, etc needs a line number - but a program which used only the absolute minimum of line numbers would be impossibly unwieldy. Line numbers should be conserved, perhaps, but only to a point of common sense. It would seem, at the least, that all code belonging to a single line number should be performing a single function, or at most, a small group of closely-related functions. I would think a good example of the latter might be the machination necessary to ask for a file name, INPUT it, do the usual cvt$$ stuff, and finally, do the OPEN.

2.0 COSMETIC ASPECTS

2.1 White Space

The DEC document makes less reference than I might like to 'white space', the use of formatting on the page for readability. There are occasions when real estate is cheap, and a few square inches in printing a large program listing is one of them. This should go for both an extra line feed or two at the end of a 'paragraph', or block of code, and for a page eject before starting a new major division of a program, such as 'Main Loop' or 'Error Handling'. Conversely, there are times when code should not be broken apart by blank lines; it's not ALWAYS mandatory that every numbered line be followed by a blank. The guiding principle should always be how to lead the reader quickly to what he wants to find by reflecting the 'shape' of the program.

2.2 Ampersands

Here's a dark, personal confession. I can't stand that damned '&' that trails every line in EXTEND mode. I'm not sure what to do about the situation, but I haven't yet managed to get so used to those dirty little marks that they don't get in the way of reading. For a while I simply used the old <Line Feed> continuation format. Then, realizing that interspersed comments were worth more than the principle of the thing, I acceded to ampersand-<CR> on each line having a comment, but still used <LF>'s to space statements. This practice (mixing formats) worries me, though - it seems like if it ever caused trouble, DEC would say it was violating rules. Now, I pretty much code things 'right', but I usually print my program listings with a program that eats ampersands. Still, I wish there were a single key which would represent the whole continuation sequence. It would

be great if <Line Feed>, say, echoed as 'ampersand - carriage return - backslash - tab'.. Drop me a line; I'll put together a patch if one or two other people are interested.

2.3 Other Cosmetics

Indenting: Personally, I like to indent all lines subsidiary to a 'FOR' by 3 columns, and all lines subsidiary to an 'IF' by 2 columns. I'm not sure that the number of columns to indent is really important enough to specify in a standard, but there are a couple of observations. First, SOME indenting should be required, because of resulting clarity in structure. Second, I personally don't like to use an extra tab, as that's indenting too far, and as deep nesting will drift right off the page. Third, whatever indenting is used should be uniform throughout a program. Fourth, comments should be indented in the same way as the lines on which they appear.

Comments: With version 6B, there is no longer any reason to not make comments appear right next to the code they describe. If you occasionally need two lines of comment for one line of code, so be it - but put the comments in line. Also, this area is one of those where I feel common sense has to apply: not every line needs a comment (heresy?). I'm thinking of 'OPEN 'KB:INPUT.CMD' FOR INPUT AS FILE 1%', or the like. If there's ANYTHING to be explained, the line gets a comment. If TRULY nothing, forget it - one line in twenty may qualify, perhaps.

Descriptive Variable Names: It's a cliche, but spell things out. This may not be a 'cosmetic aspect' of the standards, but it's at the foundation of maintainable code. In fact, when suitable names are used, a really clear line of code may not even need its (usual) comment. Also, somewhere in here I would say something about 'KISS' - Keep It Simple, Stupid. Code should be written so that someone else, guessing about what's going on, will guess correctly most of the time. Perhaps we could coin a new term, 'expectability'. This quality of program code is related to that magic quality of RSTS/E which seems to make it easy to carry most of the reference material around in your head - Martin Minow calls that 'approachibility'. In any case, KISS means that a program's flow shouldn't contain unnecessary surprises. Think about what you're writing - you KNOW what you'd expect for no-surprise code if you were going to have to read it in someone else's program. Put another way, since you always have an idea of what an experienced programmer would expect as 'natural' in reading a certain piece of your code, you can rely on his/her expectations as an 'invisible' form of documentation - IF you weed out surprising (clever) code. It often seems to boil down to trading off readability for shrinking a program by one GOSUB - that's never a tradeoff, really.

Blocks of comments: A couple of sentences of explanation at the start of each 'paragraph' of code is always an idea worth considering. A major break may justify a Form Feed and a whole narrative. Before version 6B, I followed the practice of numbering all routine entries with 'xx010' - there was a lot of 'Gosub 6010', or 'On error goto 19010' stuff. This was because the 'xx000' number was a line which contained only documentation, and introduced the routine. (It was also often preceded by a form feed.) Ease of readability required comments up front - so it was comment lines in front of the entry line. Now, however, that kind of funny business is no longer required. A paragraph can start with a natural, round number, and the first line starts with an interspersed comment for an introduction. There are no wasted line numbers, and no limit on comment length. (Hooray for 6B - it's almost worth those pesky ampersands.)

Line Format: My personal preferences don't coincide with those of DEC, so this is a case where I guess the standards shouldn't be too restrictive. (!!) Anyway, a consistent scheme should be used, at least within a given program. Personally, I like my code to look like this:

```
7000!       = = = =  Main Loop  = = = =

        ! This process steps through the ACCT.MST file

        Gosub 27500             ! Get next ACCT.MST record;
\       Goto 32500 unless MST.Record.Found%
                                ! Exit if end of file;
\       Gosub 27700             ! Catch transaction file up
```

That is, I like all of my procedure to line up. This means starting continuation lines with backslash-tab (in that order); the leftmost columns then are uniform in being line identifiers - each line begins either with a number ("I'm a new line") or with a backslash ("I'm a continuation"). Oh, well - if no one else likes this format, at least vote for a standard permissive enough to let ME keep on using it.

3.0 Comments on the DEC Standard

Here are my examples of potshots that I hope the offering of my opinions will stimulate from the SIG.

Section 1.2.1 - General Comment Lines. It looks like we no longer need lines containing only comments, destined for deletion before compile, in version 6B.

Section 1.2.2 - Block Titles. Again, there's no need to waste a line number, since comments may now precede executable code under one line number. However, code blocks SHOULD have introductions. It's not clear, though, why the exact number of tabs, etc, have to be specified; it would seem sufficient to allow any natural, narrative, and readable format.

Section 1.3.2 - Modification History. Shouldn't we make a space for authors' names as well? (People doing modifications are authors, too.) Also, although multiple lines are allowed for these categories, a single line should suffice most of the time.

Sections 1.3.3.1 and ff. See notes concerning these sections under the consideration of global, regional, and local code modules, above.

Section 1.3.3.3 Variable Definitions. This one I'm undecided about, but... can we maybe NOT describe every variable, now that we can give variables truly descriptive names? This question also depends on the availability of a cross-reference program, like the DECUS library CREF or Bloodstocks's EXREF, for instance. This is because a maintenance programmer will often need to know which variable names have been used in a particular program. Without a cross-reference facility, it would seem that the up-front list of variable names is very necessary. Also, whenever the use of a particular variable is important enough (or unusual enough), it should be documented regardless of how clear the name might be. The comments concerning global, regional, and local code modules apply here, too.

Section 1.3.6.1 - Constant Definitions. The reason for defining constants once at the beginning of a program would be mainatinability, not efficiency, in many cases. Integer constants (and floating point constants 0. and 1.) are now stored in-line in pseudo-code, and indeed are more efficient than using variables for both core and execution speed.

Section 1.4.4 - Commenting Rules, no. 1. I don't agree that each numbered line must be terminated with a blank physical line, in all cases. White space is a separator; there are times when you want to keep closely-related code close together.

Section 1.5.2.2 - Commenting Standard Routines. I would prefer to see DIM statements for standard routines put up front with other DIM statements. The only reason they would be better off within the routine is if the up-front line number allocation is not broken down with a sub-allocation for standard-routine DIM's. For maintainability, we would be better off if we could find all arrays described in one place, and to immediately know which variable names have been used.

Section 1.7.1 - Title Block. User programs should be much more liberal about keeping authors' names as part of the permanent programs listings.

Section 1.7.2 - Title Printing. I'm not sure that user program standards should require that all application programs must start up in the same way as DEC CUSP's. After all, not all users of application systems are programmers; they may be best served by interfaces designed explicitly for their needs. This is an excellent example of a situation where a programmer might want a standard on the books to refer to WHEN IT APPLIES - but which might not need to be used in every program.

Section 1.7.4 - Forbidden Usage. Thise prohibitions are meant to apply to general-usage CUSP's, distributed by the manufacturer. Obviously, application programs are not to be enjoined from 'PRINT USING', for instance.

Obviously, I don't mean criticism of the DEC document by all of the remarks above. The Digital standards were designed for a purpose different from that to which we will need to put any SIG standard. It serves us well, though, as a point of departure - and getting written material to start with is always the hardest part.

4.0 Parting Thoughts, and Things Yet to Write Up

I intend to whip up a line number skeleton.

Notes for Line Number Allocation Proposal..
Global, Regional, Local sub-allocations as discussed above, obviously
Initialization
Level-0 code (main thread)
Level-1 code (first-level subroutines)
Level-2 code (second-level subroutines)
Level-x.... (how many should we specify ?)

For I/O routines, if we include them in stds..
Bottom level - the one actually containing a GET stmt, etc
Next level up - the one calling bottom level (deblocking, etc)
xxx level (these routines are allocated inverted, ie from the bottom up)

End of Job processing needs a breakdown

Should we have levels of adherence to the standards for DECUS library and marketplace purposes ?
I'm thinking of something parallel to the "shells" of COBOL compliance.

Perhaps the kind of thing in the Programming manual's "conventions and hints" should be incorporated into the SIG standards. Incidentally, some of that material itself needs to be updated to reflect version 6Bcapabilities.

Legality of GOTO's (structured coding considerations):
a) from end of a paragraph to its beginning
(major loop control)
or is UNTIL-NEXT better ?
b) to construct IF-THEN-ELSE equivalent capability
1) forward references only
2) some kind of standard cosmetics ?
i) to make it clear it's a legal type of GOTO
ii) to make the processing flow visually clear
c) no GOTO's across pages of listing, ever